**Lab2: Combinational Logic**

Freddie Boadu

ECEN 328

Mansi Bhavsar

February 10,2019

**Introduction:** The overall objective/purpose of this lab was creating several combination logic circuits and display outputs on the device given. Using the Xilinx software, the gates we set up and demonstrated on the board. By doing AND and Or gates listed in the beginning of the lab and NOR gates toward the end, students were able to complete the design challenge at the end of the lab. Students found that word problems/real life applications could be expressed on the board using the gates learned in class.

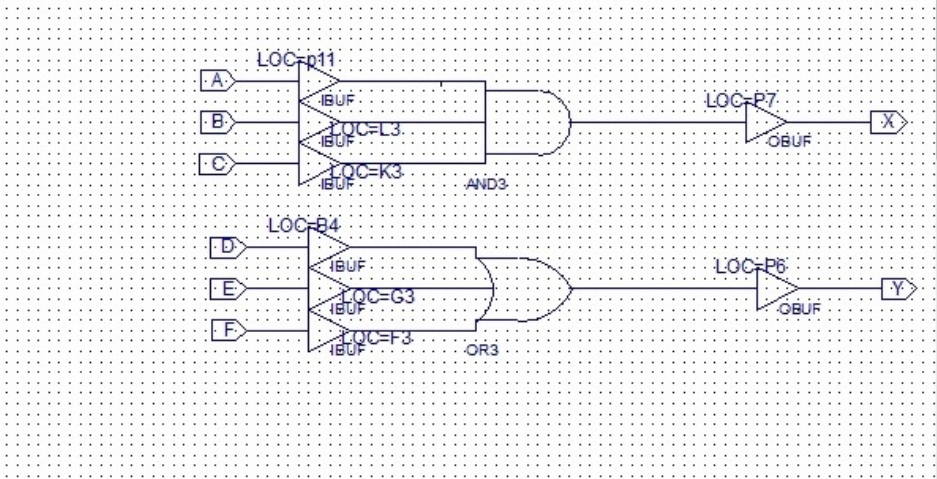
**Materials:**

* Xilinx ISE software, student or professional edition V14.7
* PC with Pentium III or higher, 128+ MB RAM and 8+ GB hard drive
* Digilent Basys2 board with an XC3S100E device.

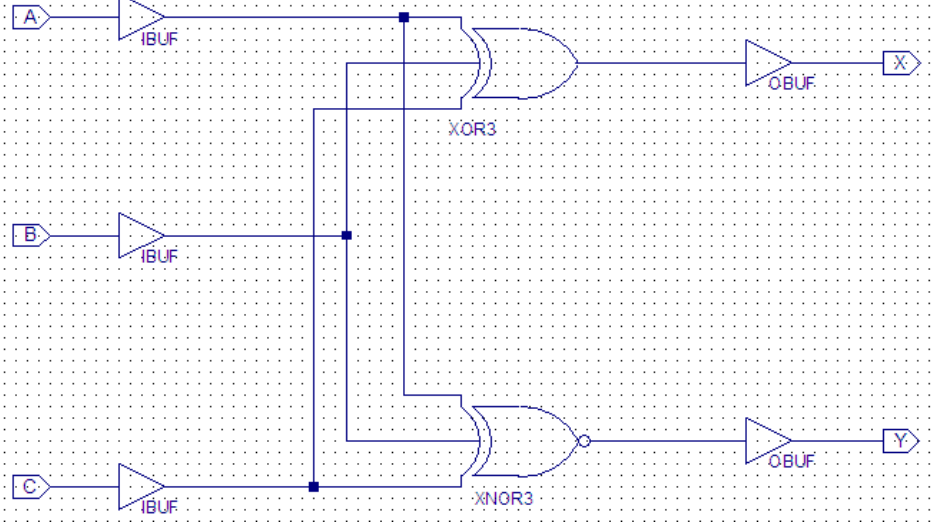
**Methods:**

This lab started with students having to assemble two basics gates. Students had to make a AND gate and a OR gate. Upon opening the program and choosing the option of schematic, A 3 input AND gate is grabbed and put onto the board. Then 3 IBUF gates are added for each input pin as well as one OBUF pin for the output. The same is done for the representation of a OR gate as well, but the 3 input AND gate is replaced with a 3 input OR gate. The switches [0:2] are used for the binary inputs for an AND gates and the switches [3:5]. A picture of the final circuit will be displayed in the DATA. For the XOR and XNOR gates, students were asked to set up the gates and demonstrate on the board. Once we set up both methods asked with the Gates and wrote down their truth tables, the students moved onto the design challenged. With the knowledge of the lab, it was determined that the design required 4 OR gates to implement everything. A truth table was filled out for this as well.

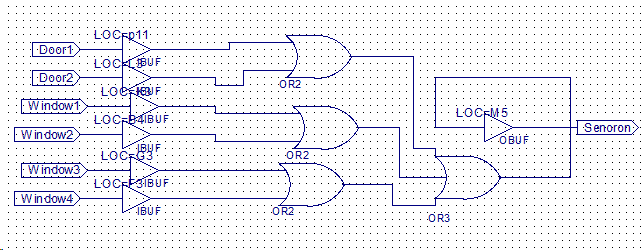
**Data:**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| A | B | C | X | Y |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 |
| 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| A | B | C | X | Y |
| 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Door 1 | Door 2 | Window 1 | Window2 | Window 3 | Window 4 | Sensor |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 |

**Results:**

The first two truth tables and schematics represent the AND OR XOR and XNOR gates respectively. The last data displayed represent the design challenge. The goal of the challenge was so showing the result of opening one door or window. When a door or window is opened the sensor will go off or is high. If all the doors and windows are closed, then the sensor is low or off. There is nothing to improve from the study.

**Conclusion:**

This lab allowed students to learn how to implement basic gates. Students also learned a real-life application of the gates during the implementation as well.